


Image encoding method and image encoding/decoding method.

Patent Number: EP0598995, B1
Publication date: 1994-06-01
Inventor(s): OKA KENICHIRO C O MITSUBISHI D (JP)
Applicant(s): MITSUBISHI ELECTRIC CORP (JP)
Requested Patent: JP6164950
Application Number: EP19930113923 19930831
Priority Number(s): JP19920316759 19921126
IPC Classification: H04N1/41
EC Classification: H04N1/41C4B, G06T9/00S
Equivalents: DE69314908D, DE69314908T, KR123790
Cited Documents: EP0261561; EP0549309; JP1188166

Abstract

An encoding method of a GBTC type that encoded data and decoded data have a fixed length, which can reduce the degree of deterioration of the quality of a reproduced image and also to provide an encoding/decoding method which, in encoding or decoding, is capable of performing editing processings to rotate an original image in 90 DEG and to rotate it inversely with respect to up and down, right and left. When encoded data in each of blocks of an original image and level specification signals ϕ_{ij} of respective pixels in each block are written into a memory or are read out therefrom, the fractions of parameter values are rounded to the nearest whole number or cut away to thereby put operation values into integral numbers, or the encoded data and the pixel level specification signal are arranged in such a manner that they can be rotated or inversely rotated an integral number of times 90 DEG . 

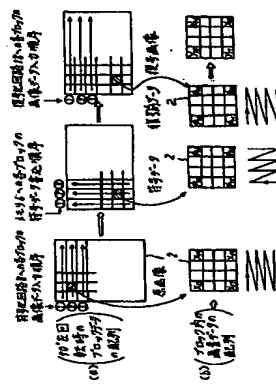
Data supplied from the esp@cenet database - I2

(51) Int. Cl. ⁸	簡別記号	庁内整理番号	FI	技術表示箇所
H04N 1/415		9070-5C		
G06F 15/66	330	C 8420-5L		
H04N 1/41	B	9070-5C		
7/13	Z			

発明者	発明者	発明者	請求項の数 3
特願 4-316759	特願 4-316759	特願 4-316759	
平成 4 年 (1992) 11 月 26 日	平成 4 年 (1992) 11 月 26 日	平成 4 年 (1992) 11 月 26 日	
(71) 出願人	000006013	000006013	(全 12 頁)
三菱電機株式会社	三菱電機株式会社	三菱電機株式会社	
東京都千代田区丸の内二丁目 2 番 3 号	東京都千代田区丸の内二丁目 2 番 3 号	東京都千代田区丸の内二丁目 2 番 3 号	
岡 賢一郎	岡 賢一郎	岡 賢一郎	
福山市緑町 1 番 8 号	福山市緑町 1 番 8 号	福山市緑町 1 番 8 号	
製作所内	製作所内	製作所内	
(72) 発明者			
三井物産株式会社	三井物産株式会社	三井物産株式会社	
東京都千代田区丸の内二丁目 2 番 3 号	東京都千代田区丸の内二丁目 2 番 3 号	東京都千代田区丸の内二丁目 2 番 3 号	
岡 賢一郎	岡 賢一郎	岡 賢一郎	
(73) 代理人			
井上 謙二	井上 謙二	井上 謙二	
井上 謙二	井上 謙二	井上 謙二	

(54) 【発明の名称】 画像符号化方法および画像符号化/復号化方法

(57) 【要約】
【目的】 符号データおよび復号化データを固定長化した GBT-C 符号化方法において、再生画像の劣化を少なくすると共に、符号化時または復号化時に、原画像の 90° 出位の回転および上下、左右を反転させる画像の行える符号化/復号化方法を得る。
【構成】 原画像 1 の各ブロック 2 の符号化データおよび各ブロック 2 の各画素のレベル指定番号 ϕ_{ij} をそれぞれメモリ 6 に書き込むか、または読み出すに際し、四捨五入または切り捨て等の処理、または符号データおよび各画素のレベル指定番号をそれぞれ 90° の整数倍の回転、反転操作を含めて配置するようにした。



【特許請求の範囲】

【請求項 1】 原画像を複数のブロックに分割して各ブロック内の各画素の階調レベルを代表する代表階調レベルをそれぞれ設定し、ブロックごとに各画素の代表階調レベルから基準レベルを算出し、各ブロック内の各画素の代表階調レベルの分布範囲を一定数に区分したレベル間隔を設定し、前記各ブロックの代表階調レベルが前記設定されたいずれかの区分に属するかを示すレベル指定番号を作成し、前記基準レベル番号、前記レベル間隔番号および前記レベル指定番号をそれぞれ独立に固定長のデータに符号化し、復号化時に上記基準レベル、レベル間隔およびレベル指定番号から復号画像データを算出するようにした画像符号化方法において、前記基準レベル、レベル間隔の算出時、および復号画像データの算出時に、増減を整理して整数値化する演算を施すことを特徴とする画像符号化方法。

【請求項 2】 原画像を複数のブロックに分割して各ブロック内の各画素の階調レベルを代表する代表階調レベルをそれぞれ設定し、ブロックごとに各画素の代表階調レベルから基準レベルを算出し、各ブロック内の各画素の代表階調レベルの分布範囲を一定数に区分したレベル間隔を設定し、前記各ブロックの代表階調レベルが前記設定されたいずれかの区分に属するかを示すレベル指定番号を作成し、前記基準レベル番号、前記レベル間隔番号および前記レベル指定番号をそれぞれ独立に固定長のデータに符号化し、復号化時に上記基準レベル、レベル間隔およびレベル指定番号から復号画像データを算出するようにしたものにおいて、符号化された各ブロックの符号データおよび各ブロック内の各画素のレベル指定番号を原画像から画素データを算出す際に、90° の整数倍の回転および反転操作を含めた配置となるような走査順序でメモリに書き込むようにしたことを特徴とする画像符号化/復号化方法。

【請求項 3】 原画像を複数のブロックに分割して各ブロック内の各画素の階調レベルを代表する代表階調レベルをそれぞれ設定し、ブロックごとに各画素の代表階調レベルから基準レベルを算出し、各ブロック内の各画素の代表階調レベルの分布範囲を一定数に区分したレベル間隔を設定し、前記各ブロックの代表階調レベルが前記設定されたいずれかの区分に属するかを示すレベル指定番号を作成し、前記基準レベル番号、前記レベル間隔番号および前記レベル指定番号をそれぞれ独立に固定長のデータに符号化し、復号化時に上記基準レベル、レベル間隔およびレベル指定番号から復号画像データを算出するようにしたものにおいて、符号化された各ブロックの符号データおよび各ブロック内の各画素のレベル指定番号が書き込まれているメモリから、復号化時に上記各ブロックの符号データおよび各ブロック内の各画素のレベル指定番号を読み出すとき、90° の整数倍の回転および反転操作を含めた配置となる走査順序で読み出すようにした。

したことを特徴とする画像符号化/復号化方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 この発明は、中間画を含む映像データの処理に適用した画像符号化方法および符号データの状態で画像復元が可能な画像符号化/復号化方法に関するものである。

【0002】

【従来の技術】 画像データを複数の画素ごとのブロックに分割し、各ブロック毎に少数の代表階調で近似させる従来のブロックトランシーション符号化方式として、例えば特開平 1-188166 号公報に示されているものがある。

【0003】 この従来の符号化方法の手順は下記の通りである。

- (1) 画像を複数の画素からなる複数のブロックに分割する。
- (2) 各ブロック毎に階調レベルの最大値 L_{max} と最小値 L_{min} の差分 $D = L_{max} - L_{min}$ を求め、以下の 3 種の符号化モードに分類する。
モード A: D が小さい場合、ブロック内を 1 レベルに符号化する。
モード B: D がやや大きい場合、ブロック内を 2 レベルに符号化する。
モード C: D が大きい場合、ブロック内を 4 レベルに符号化する。
- (3) 符号化レベルが 1 レベルの場合には平均値、2 レベルの場合には平均値より大きい画素のグループと小さいグループのそれぞれの代表値、4 レベルの場合にはブロック内の階調分布に適応した等間隔な 4 値を用いる。
- (4) 各ブロックは、ブロック内の各符号化レベルを規定する基準レベル L_a と、レベル間隔 L_d と、画素毎の各符号化レベルを規定するレベル指定番号 (2 ビット/画素) で記述される。
- (5) レベル指定番号は、ブロック間で繰返して 2 つのビットプレーン画像 (ϕ_1, ϕ_2) に変換して、それぞれ MMR (CCITT T. 5) 等の 2 値画素の符号を用いて符号化する。また、レベル間隔 L_d は可変長符号で、基準レベル L_a はブロック間差分を可変長符号で符号化する。

【0004】 また、従来の符号化方法では、符号化効率を高めるために、次のようないくつかの手段がとられていた。

- (1) 各ブロックの画素の階調レベルの最大値と最小値の差分に応じて符号化モードを 3 種類に分けて、各符号化レベル数を定めること。
- (2) レベル指定番号を 2 つのビットプレーン画像に変換して、それぞれ 2 値画素の符号化を行うこと。
- (3) レベル間隔 L_d を可変長符号化すること。

(4) 基準レベルLaは、ブロック間差分を可変長符号で符号化すること。

[0005] このようにブロック単位で符号化された可変長符号は、各ブロック毎に符号データ長が異なる。従って画像編集のために符号データ系列の中から任意のブロックを取り出すことは困難である。さらに、基準レベルLaはブロック間差分を可変長符号で符号化するの

で、任意のブロック部分の符号だけを取り出しても復号化できず、符号化時と同じ順序で、最初のブロックから順番に復号していかなければならない。

[0006] このような問題を解決する符号化方法として、「画像電子学会研究会予稿、「ハードコピー装置における画像圧縮方式の詳述」1991-04-01]

に記載されているGBTC形符号化方法がある。以下この符号化方法を説明する。

[0007] 図5は、このGBTC形符号化方法の説明図である。図5において、1は原画像、2は4×4画素の正方形で構成されるブロック、Xij(i, j=1, 2, 3, 4)は画像データ(以下「Xij」という)、φij(i, j=1, 2, 3, 4)はレベル指定番号(以下「φij」という)である。

[0008] 表1に、従来の符号データの固定長化したGBTC符号化方式のアルゴリズムを示す。

[0009]

[表1]

GBTC形符号化アルゴリズム	
P1 = (Lmax + 3Lmin) / 4	
P2 = (3Lmax + Lmin) / 4	
Q1 = mean of all xij such that xij ≤ P1	
Q4 = mean of all xij such that xij > P2	
La = (Q1 + Q4) / 2	
Ld = Q4 - Q1	
L1 = La - Ld / 4	
L2 = La + Ld / 4	
for (i = 1, ..., 4)	
for (j = 1, ..., 4)	
if xij ≤ L1	φij = 01 (binary)
else if xij ≤ La	φij = 00 (binary)
else if xij ≤ L2	φij = 10 (binary)
else	φij = 11 (binary)
end_if	
end_for	
end_for	
復号化アルゴリズム	
for (i = 1, ..., 4)	
for (j = 1, ..., 4)	
if φij = 01	yij = La - Ld / 2
else if φij = 00	yij = La - Ld / 6
else if φij = 10	yij = La + Ld / 6
else	yij = La + Ld / 2
end_if	
end_for	
end_for	

[0010] 図6は、この符号化方法を用いた信号処理系のブロック回路図で、3はホスト、4は符号化回路、5は復号化回路、6は画像データ用のメモリ(以下「メモリ」という)である。

[0011] また、図7はこの符号化回路4のブロック回路図で、7はバッファメモリ、8は符号化演算回路、9は基準レベルバッファ、10はレベル間隔バッファ、11はレベル指定番号バッファ、12は信号制御回路である。

[0012] つぎに、符号化方法について説明する。この従来の例では、画像データXijは各1バイトで構成さ

れていると仮定する。原画像1を4×4画素毎のブロック2に分割し、ブロック毎に符号化する。すなわち、まず、ブロック2の内の画素の最大値Lmaxと最小値Lminの間を4等分し、下から4分の1の値をP1、上から4分の1の値をP2とする。つぎに、Lmin以上P1以下の画素値の平均値をQ1、Lmax以下P2より大きい画素値の平均値をQ4とする。つぎにブロックの基準レベルLaを

$$La = (Q1 + Q4) / 2$$

$$Ld = Q4 - Q1$$

として求める。つぎに、レベル間隔Ldを4等分し、下から4分の1の値をL1、上から4分の1の値をL2とする。つぎに、ブロック内の16画素をそれぞれL1、L2 < xij ≤ Lmaxの場合 φij = 01 (2進値)

L1 < xij ≤ Laの場合 φij = 00 (2進値)
La < xij ≤ L2の場合 φij = 10 (2進値)
L2 < xij ≤ Lmaxの場合 φij = 11 (2進値)

【0013】 つぎに、符号化回路4の動作を、図7および図8について説明する。図7において、画像データはいったんバッファメモリに4ライン分蓄えられ、ブロック毎に符号化演算回路8に送られ、そして、既に説明したような手順でLa, Ld, φ11~φ44が求められ、それぞれ基準レベルバッファ9、レベル間隔バッファ10、レベル指定番号バッファ11を経て信号制御回路12に送られ、図8(a)に示すブロック毎の符号データとしてまとめられる。

【0014】 この符号データは図8(b)のように配置された符号系列としてメモリ6に書き込まれる。LaとLdにそれぞれ1バイト、φ11~φ44に2ビットづつ割り当てると、1ブロック分の符号データ長は6バイトになる。このように、各ブロックの符号長は6バイト※

φij = 01の場合
φij = 00の場合
φij = 10の場合
φij = 11の場合

【0018】 つぎに、復号化回路5の動作を説明する。図8において、1ブロック分の符号データは信号分配回路14でLa, Ld, φ11~φ44に分けられ、この3種類の信号はそれぞれ基準レベルバッファ15、レベル間隔バッファ16、レベル指定番号バッファ17に振り分けられ、復号化演算回路18で復号演算データyijが計算され、復号されたyijはブロック毎にバッファメモリ19にき込まれ、4ライン毎に出力される。

【0019】 発明が解決しようとする課題】従来の符号長を固定化したGBC形符号化方法は、符号化時の各パラメータP1, P2, La, Ld, L1, L2を求める演算、および復号化時の復号演算データyijを求める演算において増数が生じるため、同一の画像データに対して符号化処理と復号化処理を繰り返すような用途に用いた場合には、処理を遅らせる原因に符号化データや復号化データが変化し続け、画像として再生したときに画像が劣化し続けるという問題点があった。

【0020】 この発明は、上記のような問題を解決するためになされたもので、符号長を固定化したGBC形符号化方法において、符号化と復号化を繰り返した場合同様に再生画質が原画からの劣化が小さい範囲に留まる符号化方法を得ることを目的とする。

【0021】 また、符号化時、または復号化時に、画像を90°単位で回転させる画素または上下、左右を反転

【作用】 この発明における画像符号化方法は、符号化時、および復号化時の演算において生じる増数を、四捨五入または切り捨てることによって各パラメータおよび各ブロック内の各画素のレベル指定番号を整数化するようにしたもので、同一画像データに対して符号化と復号化を繰り返したときに、符号データおよび復号データがそ

【0025】 したものである。

【0026】 また、復号化時に、各ブロックの符号データおよび各ブロック内の各画素のレベル指定番号をメモリに格納し、復号化時に、各ブロックの符号データおよび各ブロック内の各画素のレベル指定番号をメモリから取り出すときに、それぞれ90°の整数倍の回転または反転操作を含めた処理となるように読み出すようにしたものである。

【0027】 また、復号化時に、各ブロックの符号データおよび各ブロック内の各画素のレベル指定番号をメモリから取り出すときに、それぞれ90°の整数倍の回転または反転操作を含めた処理となるように読み出すようにしたものである。

れぞれ3回目以内の符号化あるいは復号化で収まる。【0026】 また、この発明に係る画像符号化/復号化方法は、符号化時に、符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、符号化時の画像品質が向上する。

【0027】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0028】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0029】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0030】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0031】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0032】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0033】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0034】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0035】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0036】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0037】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0038】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0039】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0040】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0041】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0042】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0043】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0044】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0045】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0046】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0047】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0048】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0049】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0050】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0051】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0052】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0053】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0054】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0055】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0056】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0057】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0058】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0059】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0060】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0061】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0062】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0063】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0064】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

【0065】 また、復号化時に、ブロック毎に符号データおよび各ブロック内部のレベル指定番号を90°の整数倍の回転または反転操作を含めて配置するようにしたので、復号化時の画像品質が向上する。

実施例1 GBC形符号化アルゴリズム

P1 = (Lmax + 3Lmin) / 4 (四捨五入又は切り捨て)

P2 = (3Lmax + Lmin) / 4 (四捨五入又は切り捨て)

Q1 = mean of all xij such that xij ≤ P1 (四捨五入又は切り捨て)

Q4 = mean of all xij such that xij ≥ P2 (四捨五入又は切り捨て)

La = (Q1 + Q4) / 2 (四捨五入又は切り捨て)

Ld = Q4 - Q1 (四捨五入又は切り捨て)

L1 = La - Ld / 4 (四捨五入又は切り捨て)

L2 = La + Ld / 4 (四捨五入又は切り捨て)

for (j = 1, ..., 4)

for (i = 1, ..., 4)

if xij ≤ L1 φij = 01 (binary)

else if xij ≤ La φij = 00 (binary)

else if xij ≤ L2 φij = 10 (binary)

else φij = 11 (binary)

end_if

end_for

end_for

復号化アルゴリズム

for (j = 1, ..., 4)

for (i = 1, ..., 4)

if φij = 01 yij = La - Ld / 2 (四捨五入又は切り捨て)

else if φij = 00 yij = La - Ld / 6 (四捨五入又は切り捨て)

else if φij = 10 yij = La + Ld / 6 (四捨五入又は切り捨て)

else yij = La + Ld / 2 (四捨五入又は切り捨て)

end_if

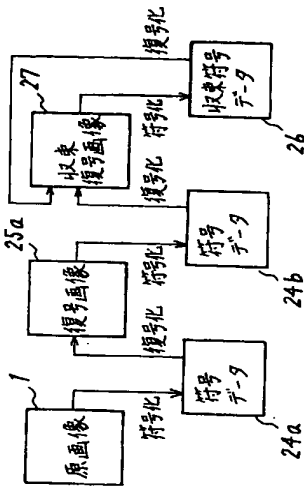
end_for

end_for

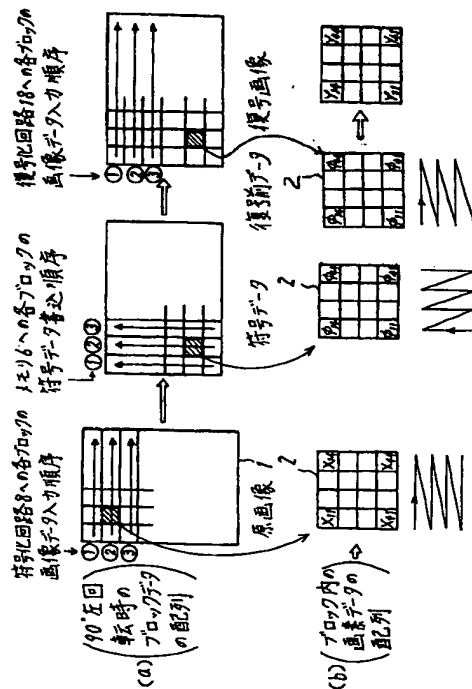
【0030】 この符号化方式のアルゴリズムと、表1に示した従来の符号化方式のアルゴリズムの相違点は、Q4をLmax以下、P2以上の画素値の平均値とするこ

と、パラメータP1, P2, Q1, Q4, La, Ld, およびyijを求める演算に増数が生じたとき、四捨五入、または切り捨てによる整数化処理を施すことであって、この

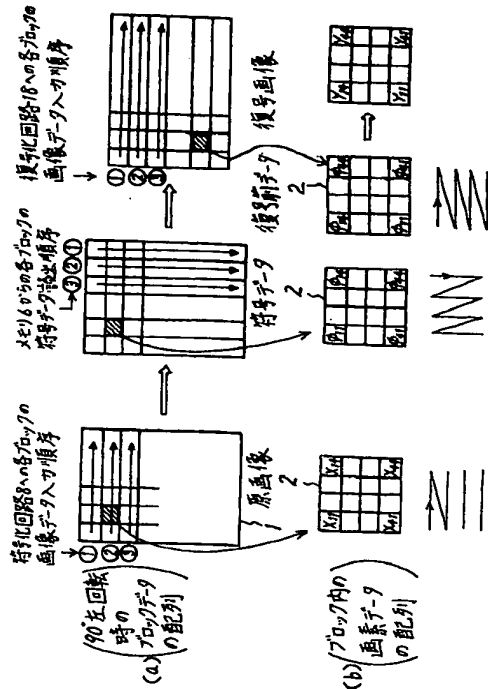
【図2】



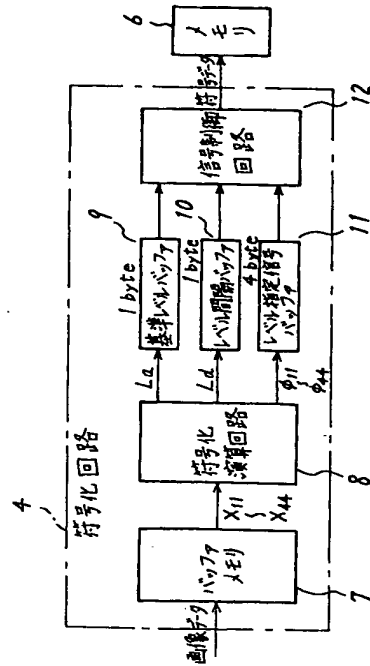
【図3】



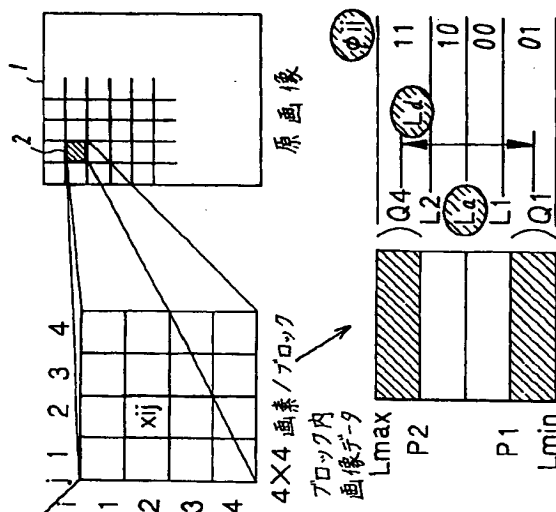
【図4】



【図7】

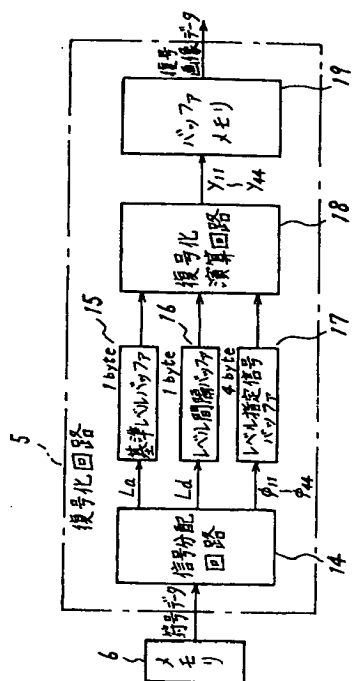


【図5】



L_{max} ブロック内の画素の最大値
 L_{min} ブロック内の画素の最小値
 P_1 $(L_{max}-L_{min})$ を4等分した
下から $1/4$ の値
 P_2 同じく上から $1/4$ の値
 Q_1 L_{min} 以上、 P_1 以下の画素値の平均値
 Q_4 L_{max} 以下、 P_2 以上の画素値の平均値
 L_a 基準レベル $= (Q_4 - Q_1) / 2$
 L_d レベル間隔 $= Q_4 - Q_1$
 L_1 L_d を4等分した下から $1/4$ の値
 L_2 同じく上から $1/4$ の値

【図9】



【手続補正等】

【提出日】平成5年9月10日

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】0012

【補正方法】変更

【補正内容】

【0012】 につき、符号化方法について説明する。この従来例では、画素データ X_{ij} は各1バイトで構成されていると仮定する。原画像1を 4×4 画素毎のブロック2に分割し、ブロック毎に符号化する。すなわち、まず、ブロック2の内の画素の最大値 L_{max} と最小値 L_{min} の間を4等分し、下から4分の1の値を P_1 、上

から4分の1の値を P_2 とする。つぎに、 L_{min} 以上 P_1 以下の画素値の平均値を Q_1 、 L_{max} 以下 P_2 より大きい画素値の平均値を Q_4 とする。つぎにブロックの基準レベル L_a を

$$L_a = (Q_1 + Q_4) / 2$$

として求め、レベル間隔 L_d を

$$L_d = Q_4 - Q_1$$

として求める。つぎに、レベル間隔 L_d を4等分し、下から4分の1の値を L_1 、上から4分の1の値を L_2 とする。つぎに、ブロック内の16画素をそれぞれ L_1 、 L_a 、 L_2 を閾値として下記のように4値に量子化する。

$L_{min} \leq x_{ij} \leq L_1$ の場合 $\phi_{ij} = 01$ (2進値)
 $L_1 < x_{ij} \leq L_a$ の場合 $\phi_{ij} = 00$ (2進値)
 $L_a < x_{ij} \leq L_2$ の場合 $\phi_{ij} = 10$ (3進値)
 $L_2 < x_{ij} \leq L_{max}$ の場合 $\phi_{ij} = 11$ (2進値)